A MULTI-RESOLUTION HIDDEN MARKOV MODEL FOR OPTIMAL DETECTION, TRACKING, SEPARATION, AND CLASSIFICATION OF MARINE MAMMAL VOCALIZATIONS

Brian F. Harrison and Paul M. Baggenstoss

Naval Undersea Warfare Center
Newport RI, 02841, USA
phone: (+001) 401-832-8240
email: harrison_bf@ieee.org
email: p.m.baggenstoss@ieee.org
web: www.nuwc.navy.mil/npt/csf/index.html

ABSTRACT

We employ the multi-resolution hidden Markov model (MRHMM) to develop an improved algorithm for modeling marine mammal wandering tone vocalizations (whistles). A vocalization is modeled by a series of time segments in which the signal has a constant frequency rate (chirps). Rather than using chirps of uniform length, the segments are allowed to be of variable size, thus adapting to both short rapid changes in frequency rate as well as long segments of constant rate. The method supports the goals of finding the single best segmentation or the average joint probability density function of the data over all possible segmentations, weighted by the *a priori* probability of each segmentation. The probability density function (PDF) projection theorem is used to allow likelihood comparisons in the raw data domain. Simulated data and recorded marine mammal vocalizations are used to demonstrate the technique.

I. THE PROBLEM ADDRESSED

Signal processing may be defined broadly as the art of extracting information from raw data in order make inferences about the nature of the data source. In many situations, the data consists of signals in noise and the exact nature, location in time, and duration of the signal are not known *a priori*. Since the best type of processing depends on the exact nature of the signal as well as its duration and time, which are unknown, the pursuit of the "optimal" processor requires a search over many degrees of freedom. Three problems arise: (A) a huge processing load, (B) the problem of making sense of the huge amount of information output by the processor, (C) the challenge of providing information in a consistent form. This last problem is due to the limits of classical decision theory that require a common "feature space" in which to make decisions.

The multi-resolution HMM (MRHMM) simultaneously addresses problems (B) and (C). Through a unique adaptation of the standard forward procedure of the hidden Markov model (HMM), the MRHMM addresses problem (B) by combining the likelihood function outputs of a large number of overlapping processing windows of various sizes into a single probabilistic model. Problem (C) is addressed with the probability density function (PDF) projection theorem and the class-specific method [1], which eliminates the need for a "common feature space" by allowing all models to compete using likelihood functions referenced to the raw data without the negative effects of high dimensionality. With limitations (B) and (C) eliminated, the only problem remaining is that of processing load which will be addressed in the future. In this paper, we explore how the MRHMM can be applied to the detection, estimation, and separation of marine mammal wandering tone vocalizations (whistles).

II. THE MRHMM

We assume familiarity with hidden Markov models (HMMs). A good reference is an article by Rabiner [2] from which we borrow some notation. The MRHMM was introduced by Baggenstoss in 2008 [3]. It is a generalization of the first-order HMM. This algorithm employs the PDF projection theorem [1], or PPT, to derive the raw-data log-likelihood functions of a set of analysis windows, then combines these windows in an optimal fashion to represent any input signal. The main concepts of the MRHMM are illustrated in Fig. 1 and will be explained next. The MRHMM is described in more detail in [3].

II-A Dwell Time Constraints

On the top of the figure, we see a time scale in divisions of T samples and a hypothetical signal containing a sinewave of length $12T$ samples and a noise burst of $4T$ samples. We assume that the signal has been produced by a system that transitions between any of M Markov states. In the figure, we show three such states: $m = 1$ "noise", $m = 2$ "sinewave", and $m = 3$ "noise burst". A standard HMM is allowed,

# Report Documentation Page	*Form Approved* *OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE **SEP 2008**	2. REPORT TYPE	3. DATES COVERED **00-00-2008 to 00-00-2008**

4. TITLE AND SUBTITLE **A Multi-Resolution Hidden Markov Model for Optimal Detection, Tracking, Separation, and Classification of Marine Mammal Vocalizations**	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	5d. PROJECT NUMBER
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) **Naval Undersea Warfare Center, , ,Newport,RI,02841**	8. PERFORMING ORGANIZATION REPORT NUMBER
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSOR/MONITOR’S ACRONYM(S)
	11. SPONSOR/MONITOR’S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

See also ADM002176. Presented at the MTS/IEEE Oceans 2008 Conference and Exhibition held in Quebec City, Canada on 15-18 September 2008.

14. ABSTRACT

see report

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**9**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

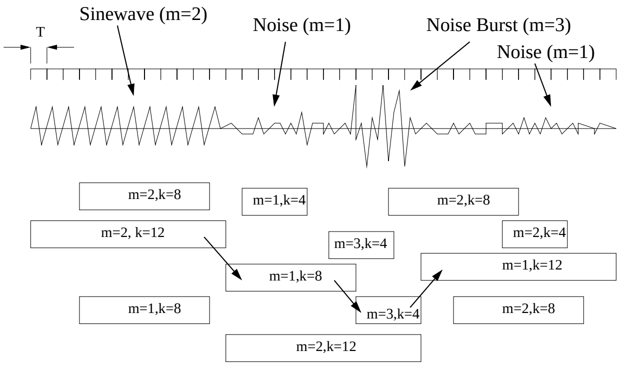

Fig. 1. Main concepts of the MRHMM.

subject to pre-determined transition probabilities, to transition from one state to another at *any time increment* and remain in a given state for *any number of time increments* (the dwell time). Unlike the standard HMM, the MRHMM puts hard constraints on the dwell times. The constraints for state m are represented by $\mathscr{K}^m$, the set of available processing window sizes in terms of T, and $\mathscr{E}^m$, the *entry* flags. We can think of the dwell time in state m as composed of a set of building blocks (or segments) selected from the set $\mathscr{K}^m$. For example, if $\mathscr{K}^m = \{12, 8, 4\}$, the dwell time in state m can be composed of any combination of segments of size $k = 12, 8$, or 4 time increments. The entry flag associated with each element of $\mathscr{K}^m$ determines if the system can enter state m with that value of k. For example, $\mathscr{E}^m = \{1, 1, 0\}$ means that the system can enter state m with either $k = 12$ or $k = 8$, but not $k = 4$. Note that this effectively means a minimum dwell time of $k = 8$ time increments.

For the hypothetical situation above, the following constraints are proposed:

State	Name	$\mathscr{K}$	$\mathscr{E}$
1	Noise	12,8,4,2	1,1,0,0
2	Sinewave	12,8,4	1,1,0
3	Noise Burst	4,2	1,1

Notice that the minimum dwell times (the smallest element of $\mathscr{K}$ with an associated entry flag of 1) are actually smaller than the segment sizes seen in Fig. 1 to allow for variations that are expected in the data.

II-B Processing Windows, Likelihood Functions, and Paths Through the State Trellis

We will now see how the dwell-time constraints translate into processing windows. In Fig. 1, under the hypothetical signal, are a set of "processing windows" or "segments" of varying length (in increments of T samples) and starting time (also in increments of T samples). Each segment is labeled with the state m and segment length k in terms of the number of time increments. Let $\mathbf{x}_{t,k}$ be the vector of data in the segment of length kT samples that starts at time increment t. Thus, it is composed of samples $1+(t-1)T$ through $(t-1+k)T$ of the input raw data. We will briefly discuss the PPT in a later section, but for now assume that we are able to calculate the raw data likelihood functions

$$L_{t,k|m} = \log p(\mathbf{x}_{t,k}|H_m),$$

where H_m is the statistical hypothesis that state m is true, for all states m, all $k \in \mathscr{K}^m$, and all t such that $\mathbf{x}_{t,k}$ is contained within the input data record. Note that each processing window in Fig. 1 corresponds to a different t, k, and m and therefore has associated with it the likelihood value $L_{t,k|m}$.

The *state trellis* is the state-vs-time plane. A potential sequence of states that the system can experience is called a *path* through the trellis. In the standard HMM, the *path* is all that must be defined in order to know the system behavior. In the MRHMM, however, it is also necessary to know the sequence of segment sizes as well. Thus, for the MRHMM, the *path* defines not only the sequence of states, but also the segment sizes (in terms of the number of time increments k).

A valid path must meet the dwell time constraints of the MRHMM. Let $\mathbf{s}$ be a valid path (a sequence of states and segment lengths) through the state trellis. The likelihood function of all data associated with a given path is written

$$L(\mathbf{X}|\mathbf{s}) = \sum_{i=1}^{n(\mathbf{s})} L_{t_i,k_i|m_i},$$

where $n(\mathbf{s})$ is the number of segments associated with path $\mathbf{s}$, and t_i, k_i, m_i are the start time, segment length, and state of the i^{th} segment in path $\mathbf{s}$. Notice that there is an implicit assumption of conditional independence among segments (conditioned on knowing the path $\mathbf{s}$).

In order to provide the necessary likelihood functions $L_{t,k|m}$, it is necessary to compute the processing windows for all valid combinations of t, k, and m. This is illustrated in Fig. 2. It is clear to see why processing load is an issue with the MRHMM. However, note that this assessment is based on the brute-force calculation of each processing window from scratch. As the windows become more heavily overlapped, the opportunity for processing reduction from time-recursive processing increases.

II-C Mathematical Problems Solved by the MRHMM

We now have the background and have made the necessary definitions in order to define the problems that we seek to solve.

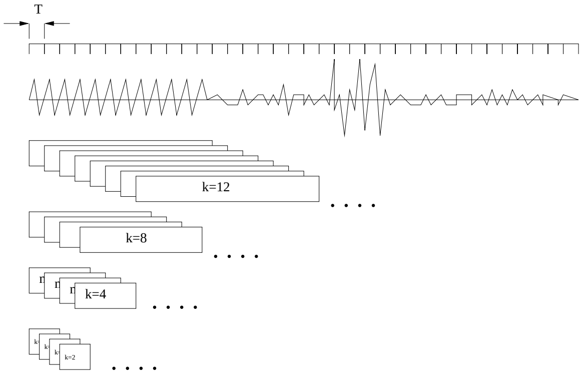

Fig. 2. Illustration of some of the processing windows that need to be computed as input to the MRHMM. In all, it is necessary to compute all window sizes at all start times for each state.

II-C.1 Average Joint PDF (Likelihood Function)

For the classification problem, it is not important which path through the state trellis had actually occurred. We are more interested in the average PDF. Let

$$p_{\text{MRHMM}}(\mathbf{X}) = \sum_{\mathbf{s}\in\mathscr{S}} p(\mathbf{s}) \, \exp\{L(\mathbf{X}|\mathbf{s})\}, \qquad (1)$$

where $\mathscr{S}$ is the set of all valid paths through the state trellis. This quantity is the average joint PDF of the raw data averaged over all valid paths and weighted by the *a priori* probability of a given path $p(\mathbf{s})$. In the same fashion as the standard first-order HMM, the *a priori* path probability is the product of the transition probabilities of the state transitions that make up the given path, which are derived from training data. All such paths are, naturally, subject to the dwell-time constraints of the MRHMM.

The summation in (1) does not need to be explicitly enumerated. The MRHMM algorithm is a modification of the HMM that is able to compute $p(\mathbf{X})$ efficiently using the *forward procedure* of the standard HMM [2] by expanding the states to include *wait states* and *slave states* and by the innovation of using *partial PDF values* [3].

II-C.2 MRHMM Initialization and Parameter Re-Estimation

The MRHMM parameter set includes the state feature PDF estimates $\hat{p}(\mathbf{z}_m|H_m)$, and the state transition probabilities. To initialize the MRHMM, training events are identified by hand-labeling a portion of data with the state labels. Features from the labeled segments can then be used to initialize the state feature PDF estimates. The *Baum-Welch* algorithm, which is used to iteratively improve the estimates of the HMM parameters, is suitably modified for the MRHMM [3]. Flat (equi-probable) state transition probabilities can be used for the first iteration of the Baum-Welch algorithm. As in the standard first-order HMM, the state feature PDF estimates are re-estimated based on the data input to each processing window, weighted by the *a posteriori* state probabilities. The state transition probabilities are also updated based on the transitions that are estimated to have occurred [3].

II-C.3 A Posteriori State Probabilities

The Baum-Welch algorithm produces, as a by-product, the *a posteriori* state probabilities. The quantity $\gamma_{t,m}$ is the probability that state m is in effect at time increment t given all the data $\mathbf{X}$. These probabilities tell the entire story about what can be inferred about the system's state and segment sequence.

II-C.4 Segmentation (Most Likely Path)

It is often important to know the single most likely path through the state trellis. This most likely path is also a segmentation of the raw data because it defines the segment sizes (and their states) that best fits the given data. Determining the most likely path requires an implementation of the Viterbi algorithm [2], although it is often the same answer obtained by maximizing $\gamma_{t,m}$ over m at each time step.

II-D MRHMM Implementation

The MRHMM is implemented by the standard forward procedure applied to an expanded set of states. In the expanded state trellis, there is a partition of k *wait states* (an artificial state for which the probability is 100 percent that the state increments by 1 on the next time step) for each segment size alloted to each state. In the preceding example, there were 9 partitions consisting of a total of $N_{\text{wait}} = 56$ wait states. The expanded state transition matrix is $N_{\text{wait}} \times N_{\text{wait}}$ in size and implements all of the dwell-time constraints of the MRHMM. The MRHMM effectively "fools" the standard forward procedure into thinking it is implementing a regular HMM with N_{wait} states. Thus, it requires a state observation probability for each state at each time step. To resolve the problem associated with multiple segment lengths, the MRHMM provides to the forward procedure the *average* log-probability of the partition, that is the log-probability of the segment divided by the segment length k. Because of the forced forward march of wait-states, the MRHMM is forced to accumulate the k partial probabilities into the full segment log-likelihood.

II-E The PDF Projection Theorem (PPT)

We have previously defined $L_{t,k|m}$ in terms of the segment raw data PDF $p(\mathbf{x}_{t,k}|H_m)$ which we will now calculate using the PDF projection theorem [1]. We describe the method briefly; greater detail can be found in the tutorial article [4]. Let $\mathbf{x}$ be a general segment of raw time-series data. Let $\mathbf{z}_m = T_m(\mathbf{x})$ be a feature set calculated from $\mathbf{x}$ specifically designed for state m. Let $\hat{p}(\mathbf{z}_m|H_m)$ be a PDF estimate of the feature set $\mathbf{z}_m$ based on training data from state m. The feature likelihood function is *projected* from the feature space to the raw data by pre-multiplying by the J-function as follows:

$$p(\mathbf{x}|H_m) = J(\mathbf{x}; T_m, H_{0,m})\ \hat{p}(\mathbf{z}_m|H_m). \qquad (2)$$

The function $p(\mathbf{x}|H_m)$ can be regarded as a function only of $\mathbf{x}$ by substituting $T_m(\mathbf{x})$ for $\mathbf{z}_m$ and can be shown to integrate to 1 over $\mathbf{x}$ (thus it is a PDF). The J-function is a unique function of $\mathbf{x}$ determined precisely from the feature transformation T_m and the class-dependent reference hypothesis $H_{0,m}$:

$$J(\mathbf{x}; T_m, H_{0,m}) = \frac{p(\mathbf{x}|H_{0,m})}{p(\mathbf{z}_m|H_{0,m})}. \qquad (3)$$

Since $J(\mathbf{x}; T_m, H_{0,m})$ is determined *a priori* without regard to training data, it can be considered the *untrained* part of $p(\mathbf{x}|H_m)$, while $\hat{p}(\mathbf{z}_m|H_m)$ is the trained part.

While it is true that $p(\mathbf{x}|H_m)$ computed in this manner is a PDF, it is only an estimate of $p(\mathbf{x}|H_m)$. The degree to which $p(\mathbf{x}|H_m)$ is a good estimate depends on (a) the accuracy of $\hat{p}(\mathbf{z}_m|H_m)$ and (b) the degree to which $\mathbf{z}_m$ is a *sufficient statistic* for the binary hypothesis test between H_m and $H_{0,m}$. In the rare case that $\mathbf{z}_m$ is in fact a sufficient statistic, the accuracy of $p(\mathbf{x}|H_m)$ depends only upon the accuracy of the low-dimensional PDF estimate $\hat{p}(\mathbf{z}_m|H_m)$.

II-E.1 Maximum Likelihood Form

The J-function takes many forms [1], one of which can be used when $\mathbf{z}_m$ are maximum likelihood (ML) estimates of a set of parameters:

$$\mathbf{z}_m = \hat{\theta}.$$

In this case, $J(\mathbf{x}; T_m, H_{0,m})$ has a simple form based on the Fisher's information matrix [1]. We have

$$p(\mathbf{x}|H_m) = \frac{p(\mathbf{x};\hat{\theta})}{(2\pi)^{-\frac{D}{2}}\ |\mathbf{I}(\hat{\theta})|^{\frac{1}{2}}}\ p(\hat{\theta}|s), \qquad (4)$$

where D is the dimension of $\hat{\theta}$ and $\mathbf{I}(\theta)$ is the Fisher's Information matrix [5]. We will utilize this form later in the development of the chirp model.

II-F MRHMM Example Using Simulated Data

To illustrate the concepts, and give examples of the output of the MRHMM, we borrow a simulated data example from a previous publication. See [3] for additional details. The signal consists of independent identically distributed (iid) Gaussian noise to which was added a low frequency (LF) pulse of autoregressive (AR) process of 128 samples in length with a peak frequency response of 0.4 radians per sample, followed by a random-length gap of at least 256 samples, followed by high frequency (HF) pulse of AR process of 64 samples with a peak frequency response of 1.2 radians per sample. An example of the spectrogram of the signal and noise is shown in Fig. 3. We implemented the MRHMM with three signal states, “Noise”, “LF pulse”, and “HF pulse”. The time increment was $T = 32$ samples, thus the signal durations are $k = 4$ and $k = 2$ for the LF and HF pulses. The dwell time constraints are listed below:

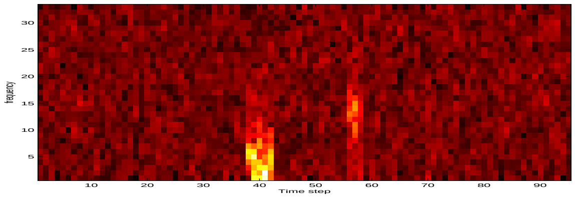

Fig. 3. Example of spectrogram of synthetic data. The data consists of three signal classes. Class 1 (noise) occurs first, then a low-frequency pulse of duration 128 samples, then noise, then a high-frequency pulse of duration 64 samples.

State	Name	$\mathscr{K}$	$\mathscr{E}$
1	Noise	8,4,2,1	1,1,1,0
2	LF Pulse	4,2,1	1,0,0
3	HF Pulse	2,1	1,0

For features, we used autoregressive linear predictive coding (LPC) features with model order P depending on the segment length k. A separate feature processor was used for each combination of k and P. Features were shared between states that had the same k and P values. Features were extracted from each analysis window by first taking the FFT, computing the magnitude squared, then computing the inverse-FFT to produce the autocorrelation function (ACF). The Levinson algorithm was used to produce the reflection coefficients of order P from the ACF. The total power in each window is also stored as the $P+1$st feature. The J-function [1] is obtained by use of the saddle-point approximation [6]. Further details on the implementation of the AR models can be found in [4].

In order to assist in intuitive understanding of the MRHMM behavior, it is useful to describe the wait states and the partial PDF matrix which are at the core of operation of the MRHMM. As we explained, to implement the dwell time restrictions, the MRHMM defines *wait states* which are artificial states that count the time increments that the system spends in a given state. For each processing window size k and state m, there is a *partition* of k wait states. In Fig. 4, we see the partial PDF matrix corresponding to the data sample in Fig. 3. The partial PDF matrix is nothing more than the segment log-likelihood values $L_{t,k|m}$ expanded along diagonal lines to fill out the wait states. As can

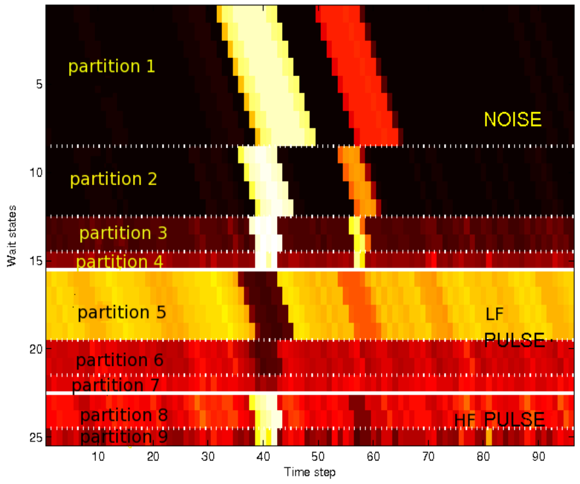

Fig. 4. Partial PDF matrix showing divisions between signal classes (solid horizontal lines) and between wait state partitions (dotted lines). Higher probability is darker.

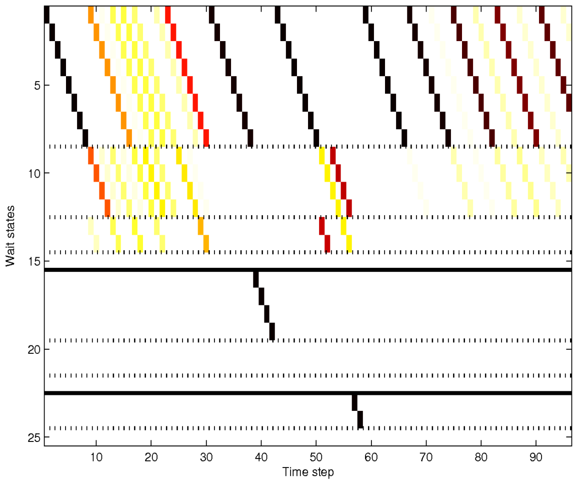

Fig. 5. Gamma probabilities expanded to include wait states. High probability is darker. The axes are the same as the partial PDF matrix in Fig. 4. In fact, the best path through Fig. 4 can be inferred from the *a posteriori* state probabilities, $\gamma_{t,m}$.

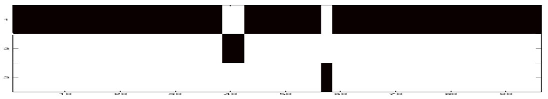

Fig. 6. Signal class probabilities calculated by summing Fig. 5 over the wait states of each class. Darker is higher probability. Time runs from left to right. Signal class identity is on the vertical axis: top = Noise, middle = LF pulse, bottom = HF pulse.

be seen on the Y axis, there are partitions of wait-states corresponding to each signal state m and each element of $\mathscr{K}^m$. Divisions between signal states are solid horizontal lines and divisions between wait state partitions are dotted lines. The log-likelihood values are scaled by $1/k$ so that if one adds up the k values along any diagonal line within a partition in the partial PDF matrix, one obtains $L_{t,k|m}$.

One way to interpret the figure is to imagine that your task is to traverse the figure and collect as much log-likelihood as possible. To do this, select a partition, then follow the diagonal lines from the top of a partition to the bottom. Once you reach the bottom of the partition, switch vertically to a new partition. Continue the process until you reach the last time increment. Whenever you switch to a new partition, you must do so in accordance with the dwell-time restrictions and the state transition probabilities.

The *a posteriori* state probabilities $\gamma_{t,m}$ computed by the Baum Welch algorithm [2] indicate the probability of each wait state at each time step given all of the data. They are obtained from the forward procedure and the backward procedure which, as the name suggests, is the forward procedure running backward in time. The values of $\gamma_{t,m}$ corresponding to Fig. 4 are shown in Fig. 5. In this figure, one can follow the trajectory that picks up the highest probabilities while meeting the dwell time restrictions. However, because all paths are represented, competing solutions can also be seen. Note that the existence of the "LF pulse" (time steps 39 through 42, partition 5, wait states 16 through 19) is clearly seen with no ambiguity. The same is true of the "HF pulse" (time steps 57 and 58, partition 8, wait states 23 and 24). At other times, it is possible to see various competing paths through the trellis. Note for example in time steps 43 through 56, the gap between the two pulses, the MRHMM is in the noise signal class. In steps 43-50, it is in partition 1, (wait states 1 through 8). Then after exiting wait state 8, it has located two possibilities to span the six time steps remaining before HF pulse occurs. It can either go into partition 2 (wait states 9 through 12), then partition 3 (wait states 13 through 14), or it can choose the reverse, partition 3 then partition 2. In this case, the higher likelihood path was partition 3 then partition 2.

The *a posteriori* state probabilities can be collapsed to indicate just the signal classes, as shown in Fig. 6. The class probabilities (Fig. 6) are an accurate indication of the true content of the data to a time resolution of $T = 32$ samples.

III. THE CHIRP MODEL

A class-specific (CS) module for chirp signals was developed to compute features from wandering narrowband frequency lines (whistles). The whistles are modeled as a sequence of linear FM chirps. The front-end processing must estimate the parameters of the chirp signal (frequency, frequency-rate, phase, noise variance) in each processing window (at every segment size and for every staring time). It is the task of the MRHMM to make sense of the myriad segment outputs. The hope is that the MRHMM will model the whistle with short segments at times where the frequency-rate was changing rapidly, and with long segments at times where it was stable. Below is the description of the processing that must be repeated for each processing window.

To apply the PPT, we used the ML form of the J-function (section II-E.1). This requires a parametric model for the data. A model for a chirp signal of length N is

$$c(n) = w(n,\theta) + \eta(n), \quad n = -\frac{N}{2}+1,\ldots,\frac{N}{2}, \qquad (5)$$

with

$$w(n,\theta) = \mathrm{Re}\left\{(a_1 + ja_2)\exp\left(j(2\pi fn + \frac{\gamma}{N}n^2)\right)\right\}, \qquad (6)$$

where $a_1 + ja_2$ is the complex amplitude, f is the start frequency, γ is the chirp rate and $\eta(n)$ are samples of white Gaussian noise with unknown variance σ^2. The module computes the ML estimates of the parameters $\theta = [a_1, a_2, f, \sigma^2, \gamma]$ as features.

Maximization of the log-likelihood function is accomplished using a two-stage procedure. First, a grid search over frequency and chirp-rate space is performed to obtain initial estimates of f and γ. The observed data $x(n)$ is multiplied by a set of chirp replicas formed by varying f and γ with the result transformed to the frequency domain to produce the frequency-chirp surface

$$S(\gamma, f) = \left| \sum_{n=-\frac{N}{2}+1}^{\frac{N}{2}} x(n + \frac{N}{2})\, e^{j(2\pi fn + \frac{\gamma}{N}n^2)} \right|^2, \qquad (7)$$

which can be efficiently computed using the FFT. The values of f and γ at the peak of $S(\gamma, f)$ provide coarse estimates of these parameters. These estimates are then used to initialize the second stage of the ML search. The log-likelihood function is given by

$$\begin{aligned} \log p(\mathbf{x}|\theta) \quad = \quad & -\tfrac{N}{2}\log(2\pi\sigma^2) \\ & -\tfrac{1}{2\sigma^2}(\mathbf{x} - \mathbf{w}(\theta))^T(\mathbf{x} - \mathbf{w}(\theta)), \end{aligned} \qquad (8)$$

where $\mathbf{x}$ and $\mathbf{w}(\theta)$ are length-N vectors of observed and modeled data respectively. Maximization of the log-likelihood function is accomplished using a Newton-Raphson search initialized with the parameter set $\theta = [0, 0, f_0, \sigma_s^2, \gamma_0]$, where f_0 and γ_0 are the estimates obtained from the grid search and σ_s^2 is the sample variance computed from the observed data. Typically, a maximum of five Newton-Raphson iterations are required for convergence.

When the feature set for a CS module is a ML estimate for the parameters of the PDF of the model, as in this case, computing the J-function is straightforward. From (4), the J-function is given as

$$J(\mathbf{x}, T, H_0) = \frac{p(\mathbf{x}|\hat{\theta})}{(2\pi)^{-D/2}\left|\mathbf{I}(\hat{\theta})\right|^{1/2}}, \qquad (9)$$

where $\hat{\theta}$ are the ML estimates of the parameters, D is the dimension of $\hat{\theta}$ and $\mathbf{I}(\hat{\theta})$ is the Fisher's information matrix (FIM) evaluated at $\hat{\theta}$. The components of the FIM are given by [5]

$$\mathbf{I}_{\theta_k,\theta_i}(\theta) = -\mathbf{E}\left(\frac{\partial^2 \ln p(\mathbf{x};\theta)}{\partial\theta_k \partial\theta_i}\right). \qquad (10)$$

For the chirp module, the elements of the FIM written in compact vector form are shown in Eq. 11, where the length-N vectors are generated using

$$\begin{aligned} \mathbf{c} &= \cos(2\pi fn + \tfrac{\gamma}{N}n^2) \\ \mathbf{s} &= \sin(2\pi fn + \tfrac{\gamma}{N}n^2) \\ \mathbf{c}_f &= -2\pi n \sin(2\pi fn + \tfrac{\gamma}{N}n^2) \\ \mathbf{s}_f &= 2\pi n \cos(2\pi fn + \tfrac{\gamma}{N}n^2) \\ \mathbf{c}_\gamma &= -\tfrac{n^2}{N}\sin(2\pi fn + \tfrac{\gamma}{N}n^2) \\ \mathbf{s}_\gamma &= \tfrac{n^2}{N}\cos(2\pi fn + \tfrac{\gamma}{N}n^2) \end{aligned} \qquad (12)$$

for $n = -\frac{N}{2}+1,\ldots,\frac{N}{2}$. Operating in the log domain, the J-function for the chirp module is written as

$$\begin{aligned} \log J(\mathbf{x}, T, H_0) \quad = \quad & -\tfrac{N}{2}\log(2\pi\sigma^2) \\ & -\tfrac{1}{2\sigma^2}(\mathbf{x} - \mathbf{w}(\hat{\theta}))^T(\mathbf{x} - \mathbf{w}(\hat{\theta})) \\ & -\log\left((2\pi)^{-D/2}\left|\mathbf{I}(\hat{\theta})\right|^{1/2}\right). \end{aligned} \qquad (13)$$

IV. RESULTS

We first apply the MRHMM to a simple whistle sequence, then to a superimposed set of whistles plus clicks.

IV-A Simple Whistle Analysis

To demonstrate the MRHMM on biologic whistles, we created an MRHMM using two states: "Chirp" and "Noise". For the chirp state, we used the ML chirp model features described above. For the noise state, we used only noise power features. The time increment was $T = 64$ samples. The dwell time constraints are summarized below:

State	$\mathscr{K}$	$\mathscr{E}$	Features
Chirp	12,6,4,3,2,1	1,1,1,1,0,0	ML Chirp
Noise	12,6,4,3,2,1	1,1,1,1,0,0	Power

The largest segment size was $12 \times 64 = 768$ samples.

For illustration, we selected the whistle sequence illustrated in Fig. 7. The whistle was frequency-shifted and downsampled to 8333 Hz to occupy most of the band for clarity. The time-series and spectrogram are shown on the top two graphics in the figure. The

$$\begin{bmatrix} \frac{1}{\sigma^2}\mathbf{c}^T\mathbf{c} & -\frac{1}{\sigma^2}\mathbf{c}^T\mathbf{s} & \frac{1}{\sigma^2}(a_1\mathbf{c}_f^T\mathbf{c}-a_2\mathbf{c}^T\mathbf{s}_f) & 0 & \frac{1}{\sigma^2}(a_1\mathbf{c}_\gamma^T\mathbf{c}-a_2\mathbf{c}^T\mathbf{s}_\gamma) \\ -\frac{1}{\sigma^2}\mathbf{s}^T\mathbf{c} & \frac{1}{\sigma^2}\mathbf{s}^T\mathbf{s} & -\frac{1}{\sigma^2}(a_1\mathbf{s}^T\mathbf{c}_f-a_2\mathbf{s}_f^T\mathbf{s}) & 0 & -\frac{1}{\sigma^2}(a_1\mathbf{s}^T\mathbf{c}_\gamma-a_2\mathbf{s}^T\mathbf{s}_\gamma) \\ \frac{1}{\sigma^2}(a_1\mathbf{c}_f^T\mathbf{c}-a_2\mathbf{s}_f^T\mathbf{c}) & -\frac{1}{\sigma^2}(a_1\mathbf{c}_f^T s-a_2\mathbf{s}_f^T\mathbf{s}) & \frac{1}{\sigma^2}\left|(a_1\mathbf{c}_f-a_2\mathbf{s}_f)\right|^2 & 0 & \frac{1}{\sigma^2}\left|(-a_1\mathbf{c}_f+a_2\mathbf{s}_f)\right|^2 \\ 0 & 0 & 0 & \frac{N}{2\sigma^4} & 0 \\ \frac{1}{\sigma^2}(a_1\mathbf{c}_\gamma^T\mathbf{c}-a_2\mathbf{s}_\gamma^T\mathbf{c}) & -\frac{1}{\sigma^2}(a_1\mathbf{c}_\gamma^T s-a_2\mathbf{s}_\gamma^T\mathbf{s}) & \frac{1}{\sigma^2}\left|(-a_1\mathbf{c}_\gamma+a_2\mathbf{s}_\gamma)\right|^2 & 0 & \frac{1}{\sigma^2}\left|(a_1\mathbf{c}_\gamma-a_2\mathbf{s}_\gamma)\right|^2 \end{bmatrix} \quad (11)$$

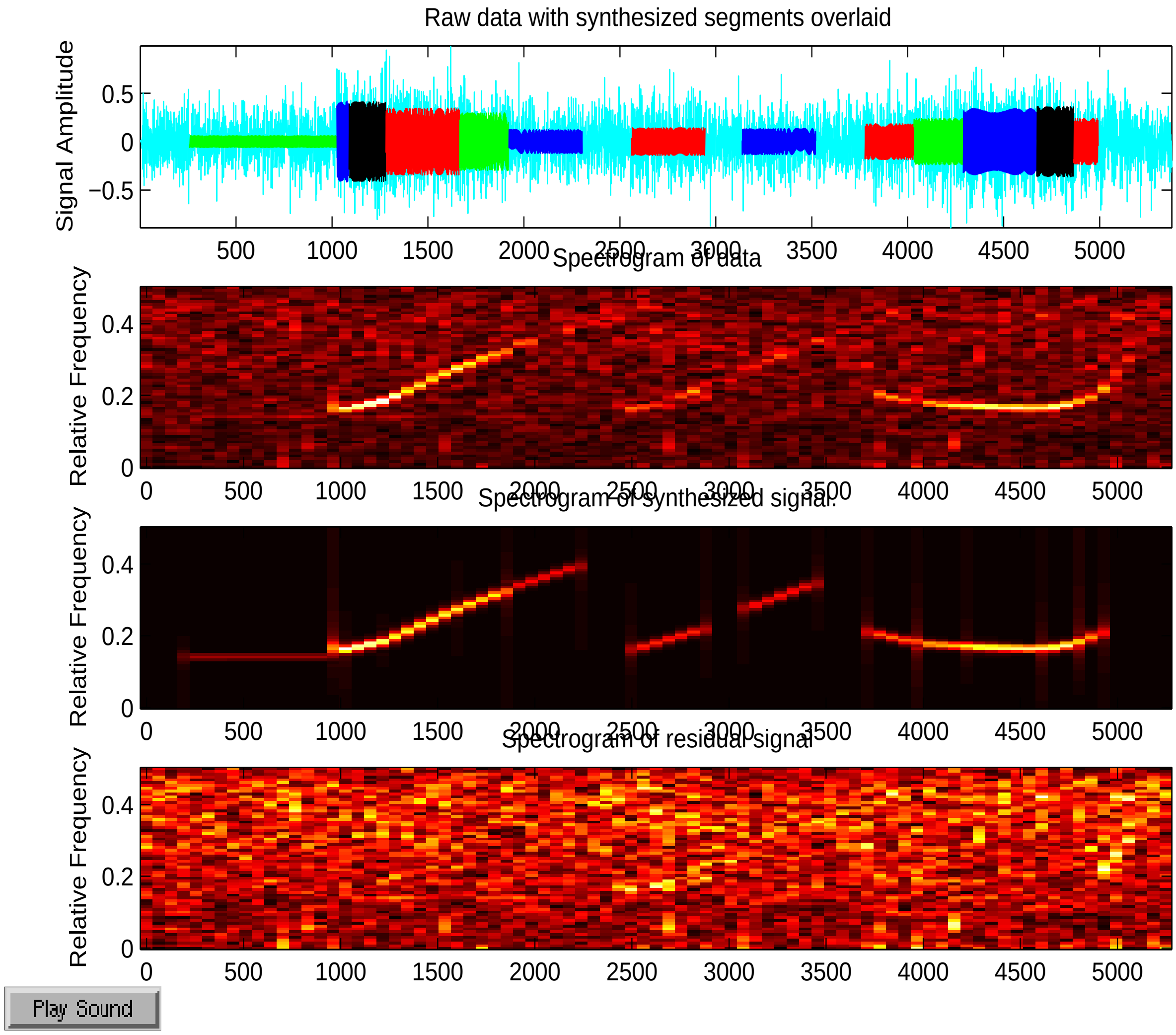

Fig. 7. Spectrograms of raw data, synthesized signal and residual.

synthetic chirp signals are overlaid in alternating colors on top of the time-series. These were synthesized from the estimated chirp parameters of the segments corresponding to the most likely segmentation (path). Those segments that are classified as “Noise” have no synthetic signal overlay. The spectrogram of the purely synthetic signal is shown on the third graphic. On the fourth (bottom) graphic, the synthetic signal is subtracted from the input data revealing the spectrogram of the residual noise. As expected, the “optimal” segmentation exhibits short segments during times of rapid change, and long segments during times of stability. Note also the classification of segments with no chirp as “Noise”. Also interesting to note is the capturing of both the weak tonal interference at the start of the time-series and the weak chirp in the middle.

IV-B Whistle Separation

To demonstrate the ability of the MRHMM to assist in signal separation, it was applied to superimposed marine mammal whistles and clicks. The spectrogram of the signal we will use to demonstrate the technique is shown in the upper plot of Fig. 8. As seen in the figure the signal consists of three components, those being two marine mammal whistles and a set of clicks. The weaker of the two whistles appears to be a multipath reflection of the stronger. All three components overlap simultaneously in time and frequency.

We implemented the MRHMM assuming two states corresponding to the signal classes: "Noise" and "Chirp". This time, the ML chirp feature module was used to compute features for both signal and noise classes. For the noise portions of the data, we would expect the ML estimate of the complex amplitude of the chirp model to have a magnitude of nearly zero. We selected an elemental segment length of $T = 128$ samples with six analysis window lengths: $[128, 256, 512, 768, 1024, 2048]$. Therefore, the dwell time constraints are:

State	$\mathscr{K}$	$\mathscr{E}$	Features
Noise	1,2,4,6,8,16	1,1,1,1,1,1	ML Chirp
Chirp	1,2,4,6,8,16	1,1,1,0,0,0	ML Chirp

Fig. 8 shows the spectrogram of the raw data and the gamma probabilities computed by the MRHMM. The partitions for the noise states are shown in the top half of the gamma probabilities plot separated by red horizontal lines and those for the chirp states are shown in green in the bottom half of the plot. Fig. 9 shows the spectrograms of the raw data, the synthesized data and the residual signal. The residual is computed by subtracting the synthesized data from the raw data. The synthesized signal is generated from (6) using the set of features from the signal class with the highest likelihood at each time step. The synthesized signal produced a highly accurate reconstruction of the high power whistle, which when subtracted from the raw data, produced a residual signal with the low power whistle and clicks clearly intact. The residual signal was again processed by the MRHMM. Fig. 10 shows the spectrograms of the residual signal, synthesized signal and second residual. The spectrogram of the second residual demonstrates the effectiveness of the technique in removing the low power whistle while leaving the clicks intact.

A potential area for improvement of the existing algorithm is illustrated near the end of the superimposed whistles (near time step 600) where the MRHMM switches between modeling the two whistles. This problem is attributed to the assumption of independence of the segments given the state index. Improved continuity would result if the prediction of

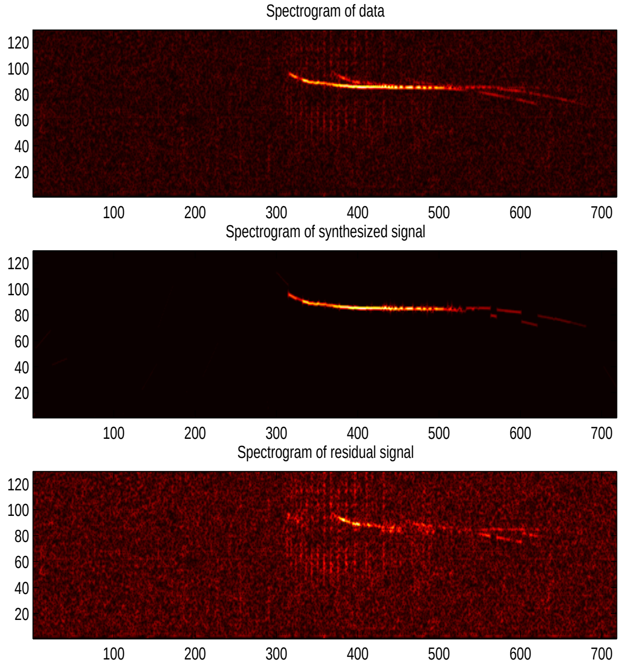

Fig. 9. Spectrograms of raw data, synthesized signal and residual.

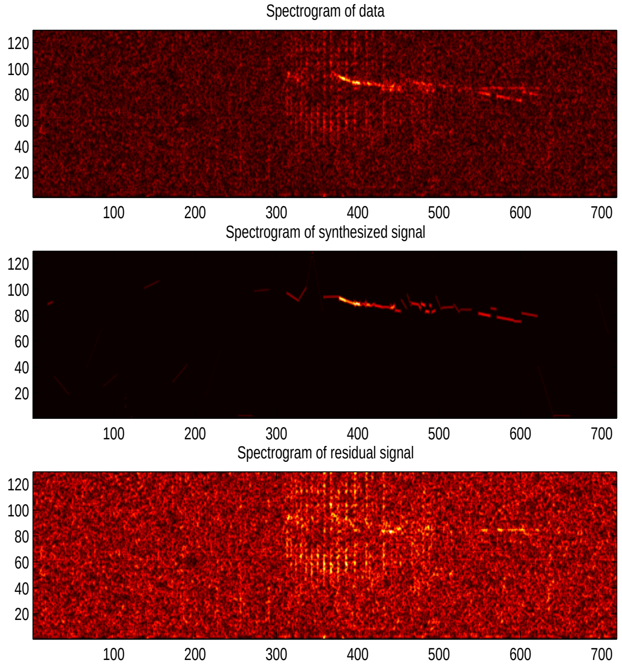

Fig. 10. Spectrograms of first residual, synthesized signal and second residual.

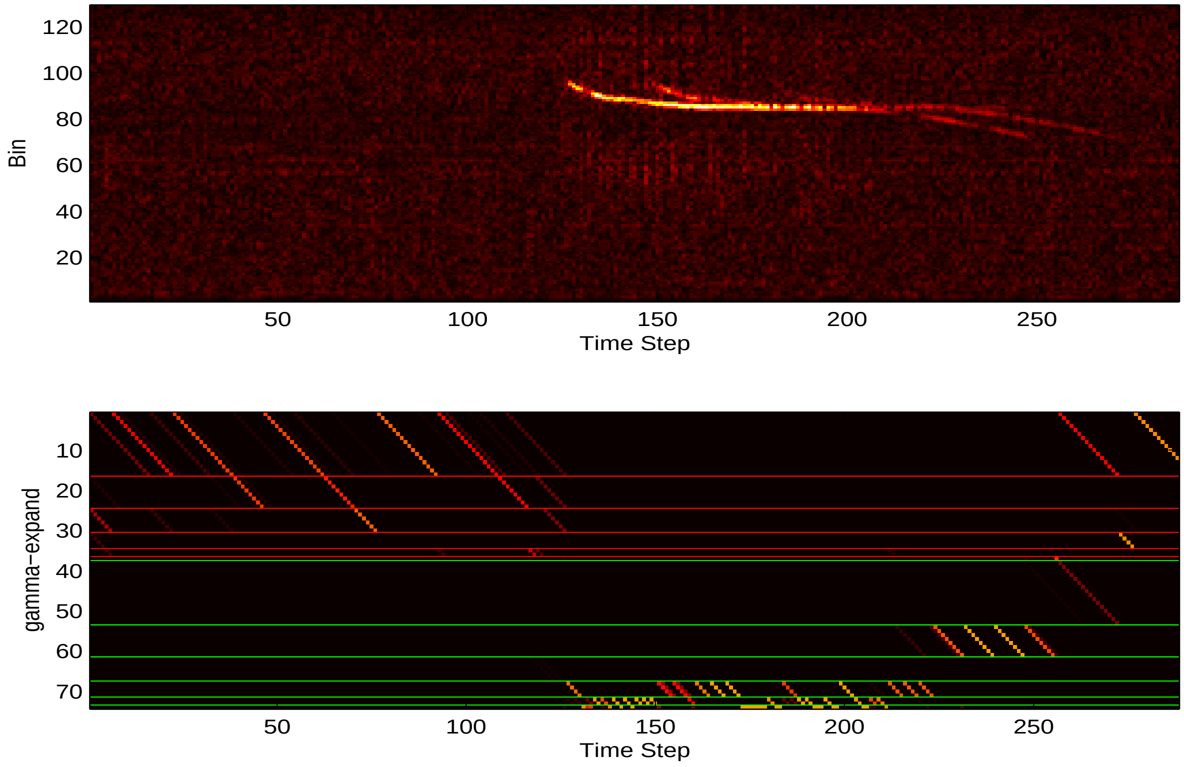

Fig. 8. Spectrogram of raw data and gamma probabilities (higher probability is lighter) computed by the MRHMM.

chirp frequency and frequency rate was integrated into the statistical model. This is the topic of future work.

V. CONCLUSIONS

We have demonstrated the power of the MRHMM as an analysis tool for modeling and separation of marine mammal vocalizations. The ability of the MRHMM to find optimal segmentations of the data into piecewise linear FM chirps produced highly accurate models that can be used to coherently remove whistles, leaving a residual signal in which additional superimposed signals are clearly evident. This technique may have applications, for example, in density estimation of marine mammals.

REFERENCES

[1] P. M. Baggenstoss, “The PDF projection theorem and the class-specific method,” *IEEE Trans Signal Processing*, pp. 672–685, March 2003.

[2] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proceedings of the IEEE*, vol. 77, pp. 257–286, February 1989.

[3] P. M. Baggenstoss, “A multi-resolution hidden markov model using class-specific features,” *Proceedings of EUSIPCO 2008, Lausanne, Switzerland*, Aug 2008.

[4] P. M. Baggenstoss, “The class-specific classifier: Avoiding the curse of dimensionality (tutorial),” *IEEE Aerospace and Electronic Systems Magazine, special Tutorial addendum*, vol. 19, pp. 37–52, January 2004.

[5] S. Kay, *Fundamentals of Statisticsl Signal Processing, Estimation Theory*. Prentice Hall, Upper Saddle River, New Jersey, USA, 1993.

[6] S. M. Kay, A. H. Nuttall, and P. M. Baggenstoss, “Multidimensional probability density function approximation for detection, classification and model order selection,” *IEEE Trans. Signal Processing*, pp. 2240–2252, Oct 2001.